FIG. 1

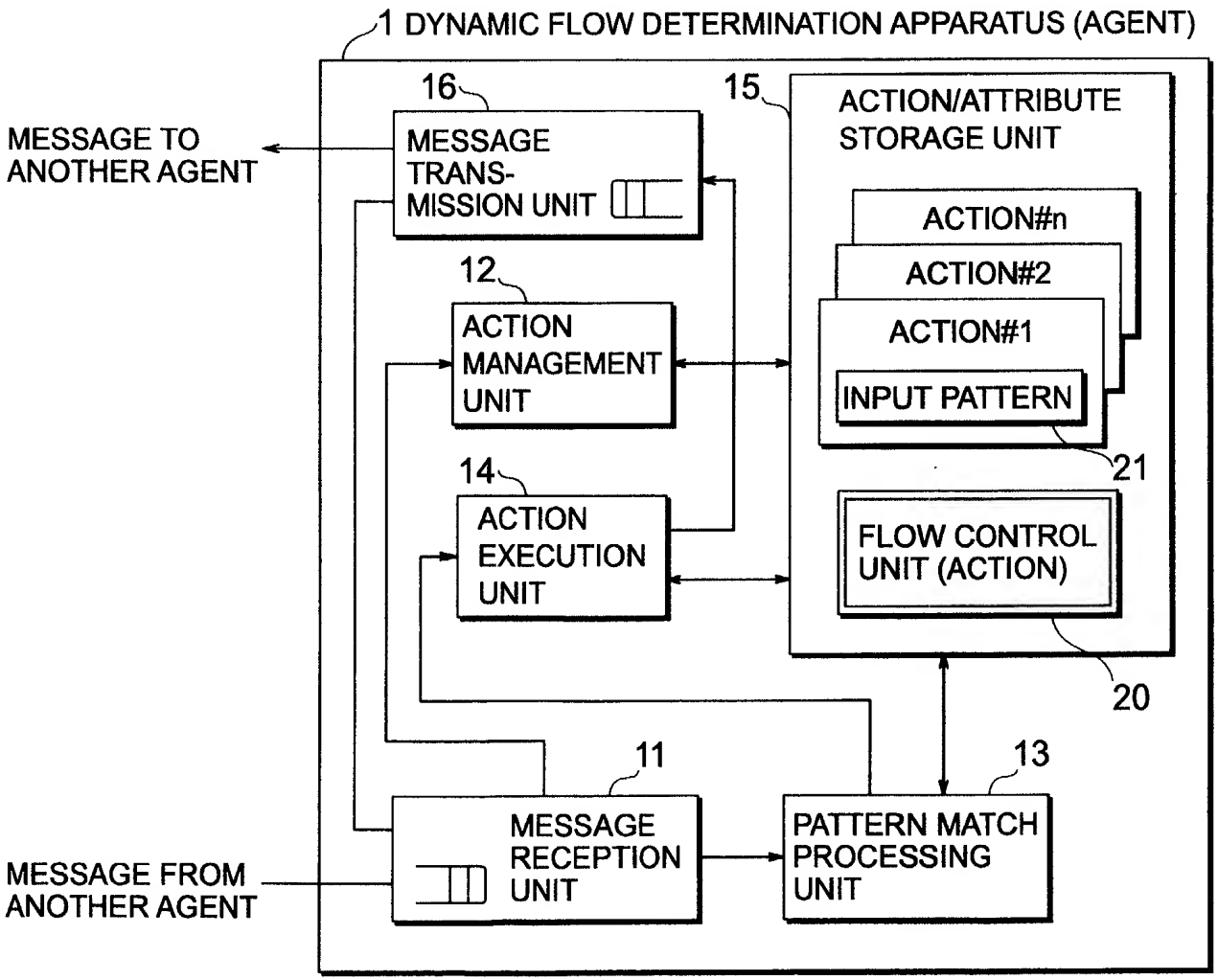
1 DYNAMIC FLOW DETERMINATION APPARATUS (AGENT)
16
MESSAGE TO ANOTHER AGENT
MESSAGE TRANS-MISSION UNIT
15
ACTION/ATTRIBUTE STORAGE UNIT
ACTION#n
ACTION#2
ACTION#1
INPUT PATTERN
21
12
ACTION MANAGEMENT UNIT
14
ACTION EXECUTION UNIT
FLOW CONTROL UNIT (ACTION)
20
11
13
MESSAGE FROM ANOTHER AGENT
MESSAGE RECEPTION UNIT
PATTERN MATCH PROCESSING UNIT

FIG. 2

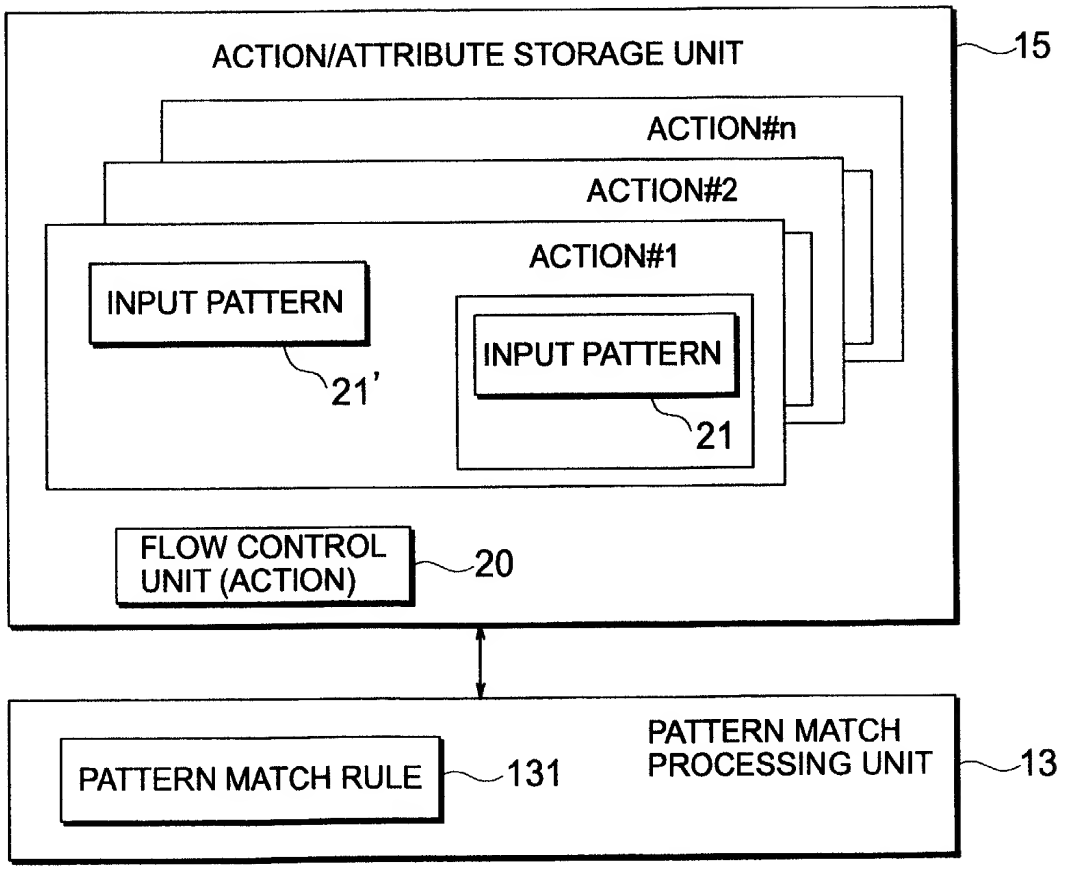
ACTION/ATTRIBUTE STORAGE UNIT
15
ACTION#n
ACTION#2
ACTION#1
INPUT PATTERN
21'
INPUT PATTERN
21
FLOW CONTROL UNIT (ACTION)
20
PATTERN MATCH RULE
131
PATTERN MATCH PROCESSING UNIT
13

FIG. 3

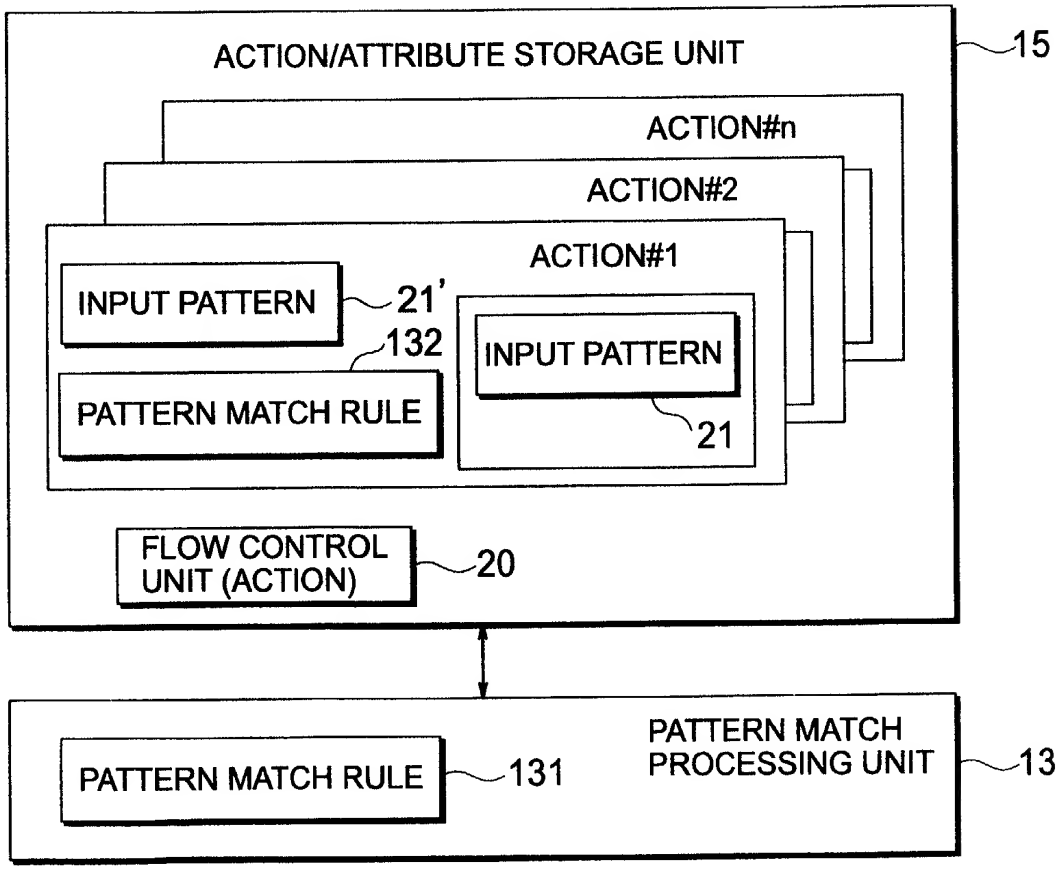
ACTION/ATTRIBUTE STORAGE UNIT
15
ACTION#n
ACTION#2
ACTION#1
INPUT PATTERN
21'
132
PATTERN MATCH RULE
INPUT PATTERN
21
FLOW CONTROL UNIT (ACTION)
20
PATTERN MATCH RULE
131
PATTERN MATCH PROCESSING UNIT
13

FIG. 4

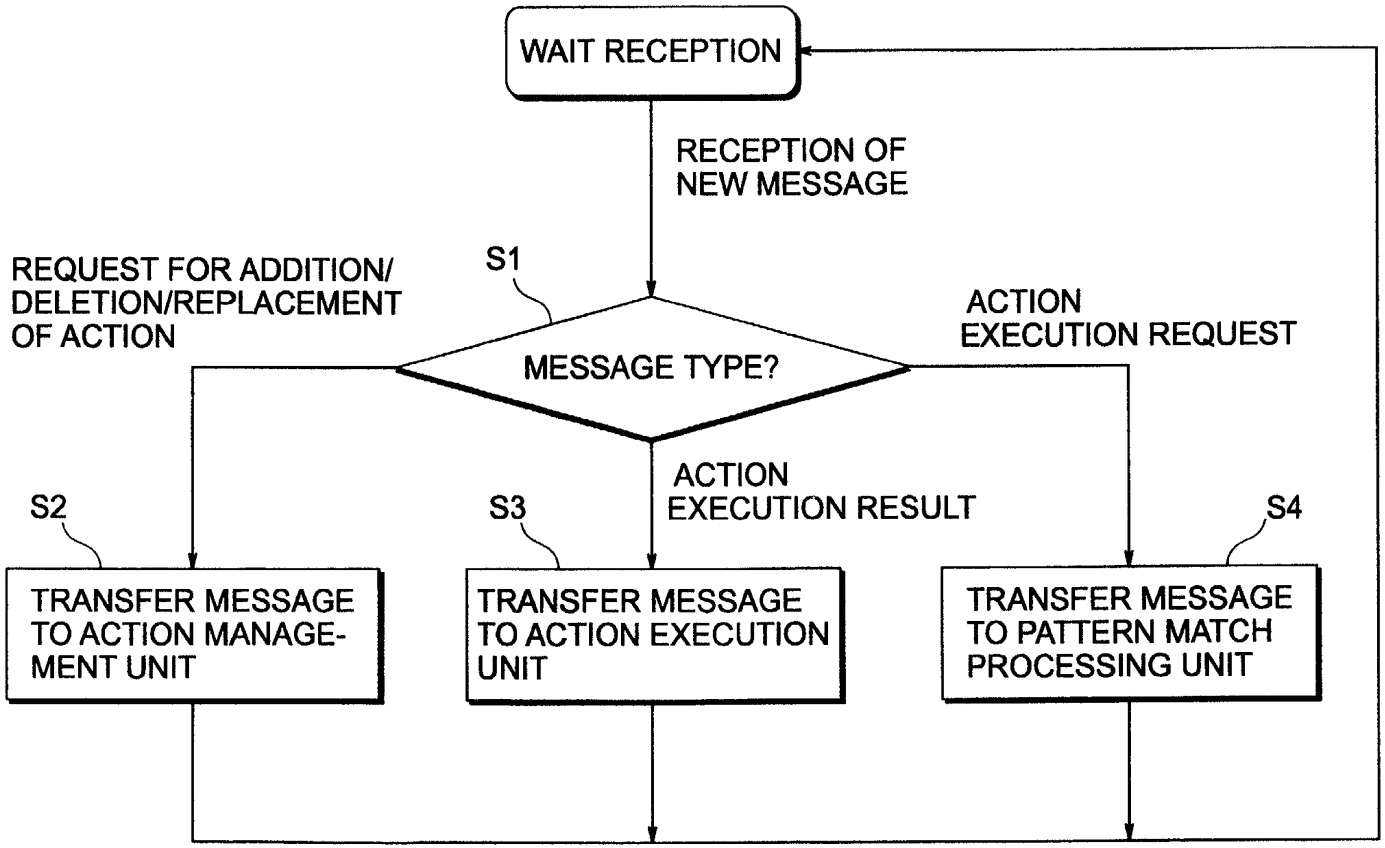
WAIT RECEPTION
RECEPTION OF NEW MESSAGE
S1
MESSAGE TYPE?
REQUEST FOR ADDITION/ DELETION/REPLACEMENT OF ACTION
ACTION EXECUTION REQUEST
ACTION EXECUTION RESULT
S2
S3
S4
TRANSFER MESSAGE TO ACTION MANAGE-MENT UNIT
TRANSFER MESSAGE TO ACTION EXECUTION UNIT
TRANSFER MESSAGE TO PATTERN MATCH PROCESSING UNIT

FIG. 5

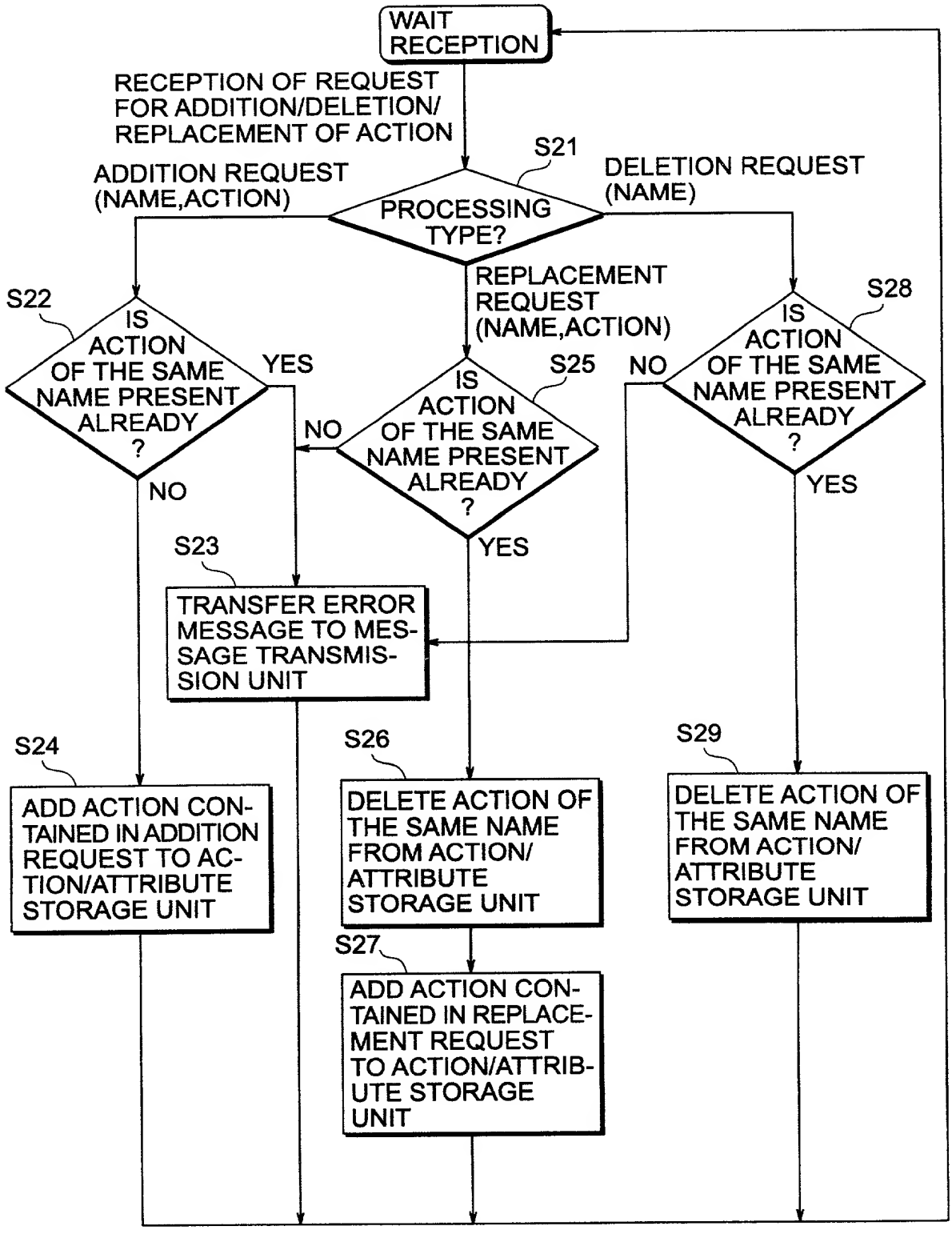
WAIT RECEPTION
RECEPTION OF REQUEST FOR ADDITION/DELETION/ REPLACEMENT OF ACTION
S21
PROCESSING TYPE?
ADDITION REQUEST (NAME,ACTION)
REPLACEMENT REQUEST (NAME,ACTION)
DELETION REQUEST (NAME)
S22
IS ACTION OF THE SAME NAME PRESENT ALREADY ?
YES
NO
S25
IS ACTION OF THE SAME NAME PRESENT ALREADY ?
NO
YES
S28
IS ACTION OF THE SAME NAME PRESENT ALREADY ?
NO
YES
S23
TRANSFER ERROR MESSAGE TO MESSAGE TRANSMISSION UNIT
S24
ADD ACTION CONTAINED IN ADDITION REQUEST TO ACTION/ATTRIBUTE STORAGE UNIT
S26
DELETE ACTION OF THE SAME NAME FROM ACTION/ ATTRIBUTE STORAGE UNIT
S27
ADD ACTION CONTAINED IN REPLACEMENT REQUEST TO ACTION/ATTRIBUTE STORAGE UNIT
S29
DELETE ACTION OF THE SAME NAME FROM ACTION/ ATTRIBUTE STORAGE UNIT

FIG. 6

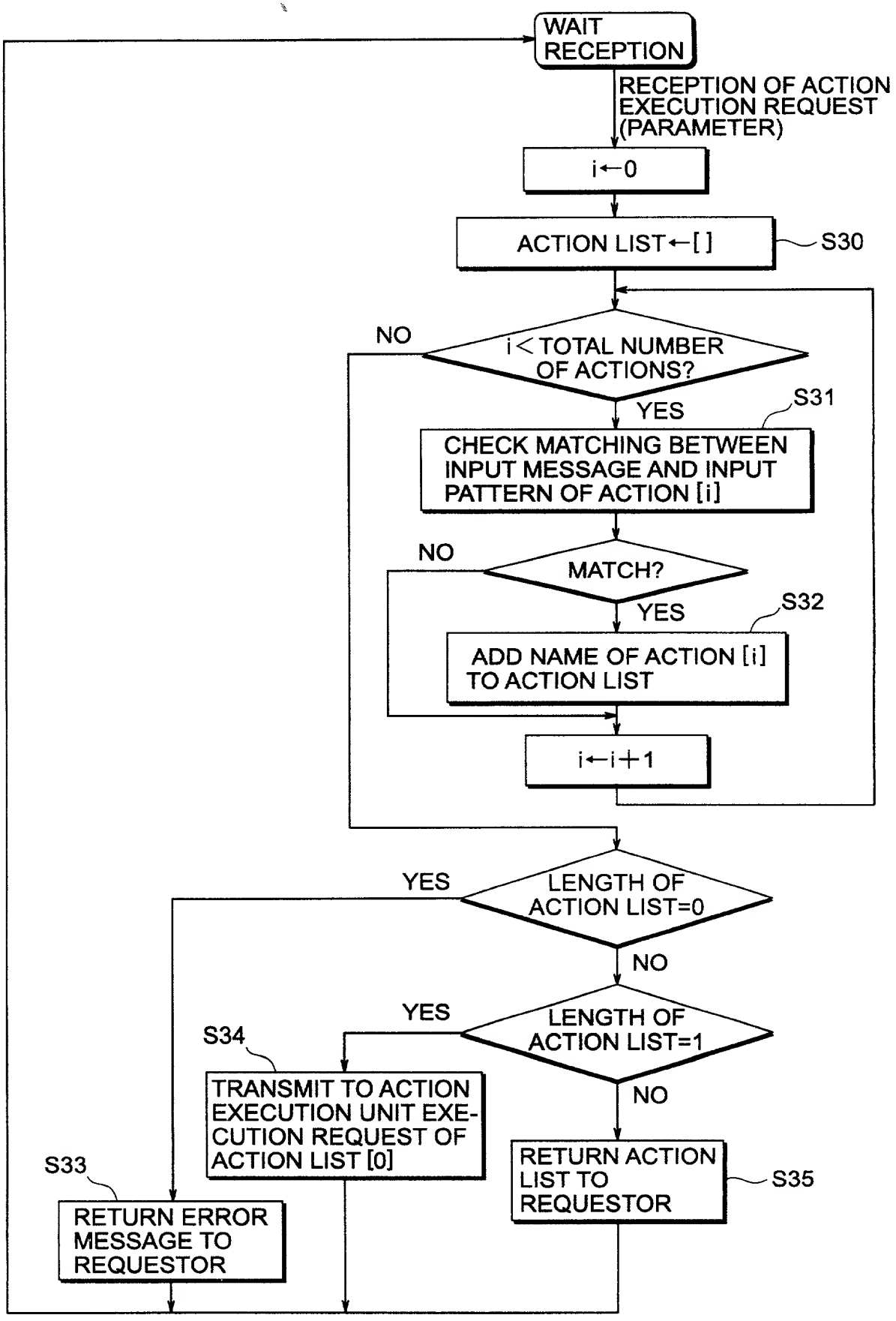
WAIT RECEPTION
RECEPTION OF ACTION EXECUTION REQUEST (PARAMETER)
i←0
ACTION LIST←[ ]
S30
NO
i < TOTAL NUMBER OF ACTIONS?
YES
S31
CHECK MATCHING BETWEEN INPUT MESSAGE AND INPUT PATTERN OF ACTION [i]
NO
MATCH?
YES
S32
ADD NAME OF ACTION [i] TO ACTION LIST
i←i+1
YES
LENGTH OF ACTION LIST=0
NO
YES
LENGTH OF ACTION LIST=1
NO
S34
TRANSMIT TO ACTION EXECUTION UNIT EXECUTION REQUEST OF ACTION LIST [0]
RETURN ACTION LIST TO REQUESTOR
S35
S33
RETURN ERROR MESSAGE TO REQUESTOR

FIG. 7

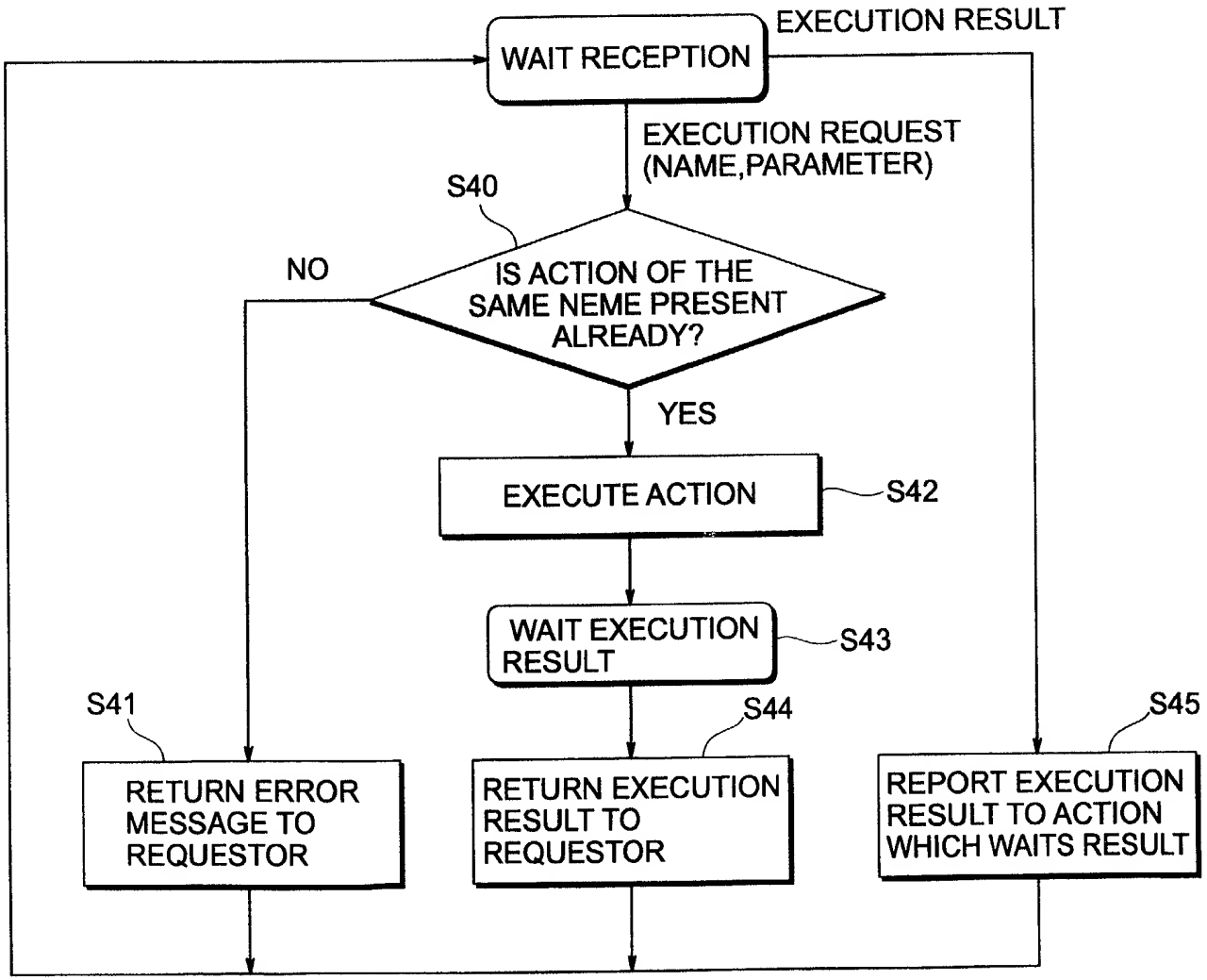
WAIT RECEPTION
EXECUTION RESULT
EXECUTION REQUEST
(NAME,PARAMETER)
S40
IS ACTION OF THE SAME NEME PRESENT ALREADY?
NO
YES
EXECUTE ACTION
S42
WAIT EXECUTION RESULT
S43
S41
S44
S45
RETURN ERROR MESSAGE TO REQUESTOR
RETURN EXECUTION RESULT TO REQUESTOR
REPORT EXECUTION RESULT TO ACTION WHICH WAITS RESULT

FIG. 8

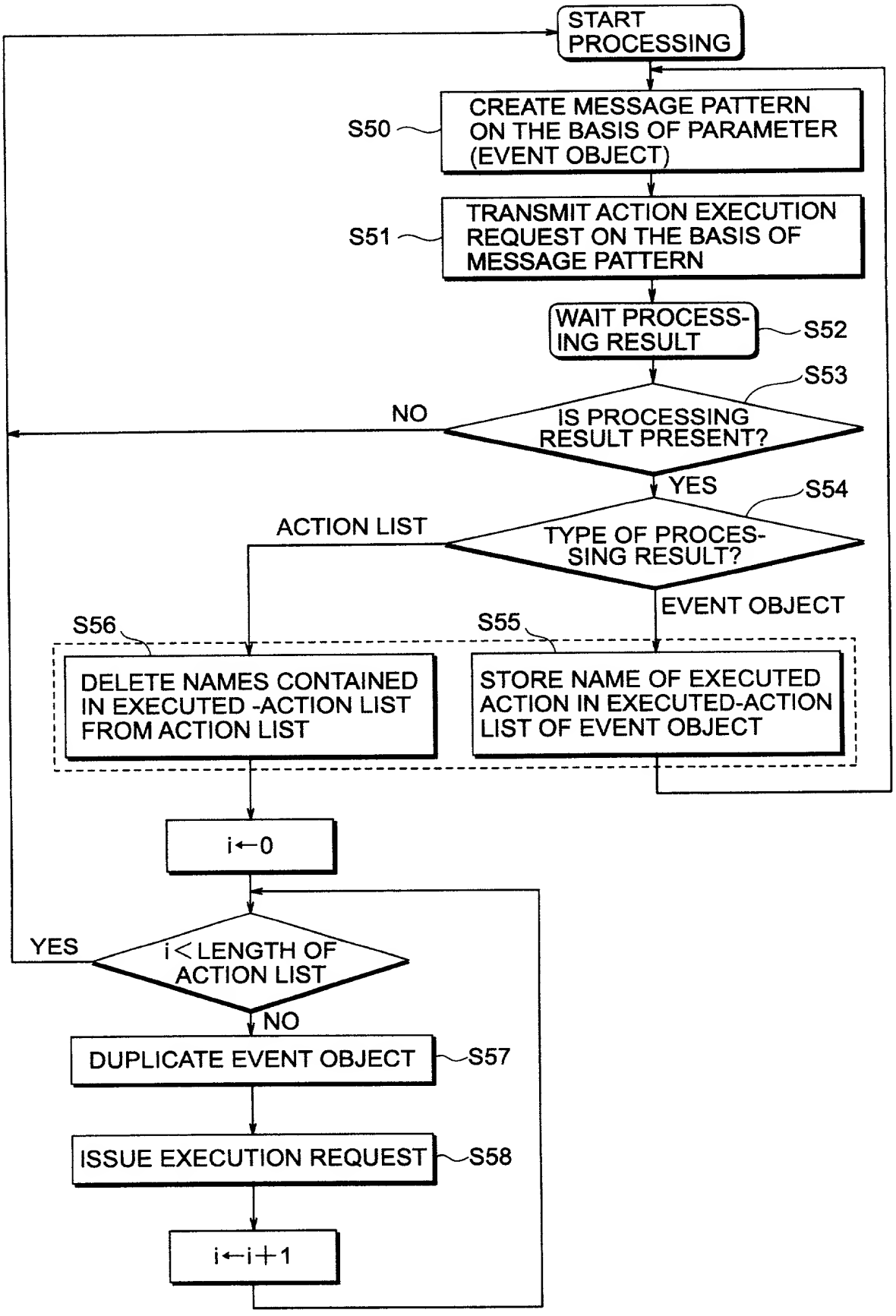
START PROCESSING
S50
CREATE MESSAGE PATTERN ON THE BASIS OF PARAMETER (EVENT OBJECT)
S51
TRANSMIT ACTION EXECUTION REQUEST ON THE BASIS OF MESSAGE PATTERN
WAIT PROCESSING RESULT
S52
S53
IS PROCESSING RESULT PRESENT?
NO
YES
S54
TYPE OF PROCESSING RESULT?
ACTION LIST
EVENT OBJECT
S56
S55
DELETE NAMES CONTAINED IN EXECUTED -ACTION LIST FROM ACTION LIST
STORE NAME OF EXECUTED ACTION IN EXECUTED-ACTION LIST OF EVENT OBJECT
i←0
YES
i<LENGTH OF ACTION LIST
NO
DUPLICATE EVENT OBJECT
S57
ISSUE EXECUTION REQUEST
S58
i←i+1

FIG. 9

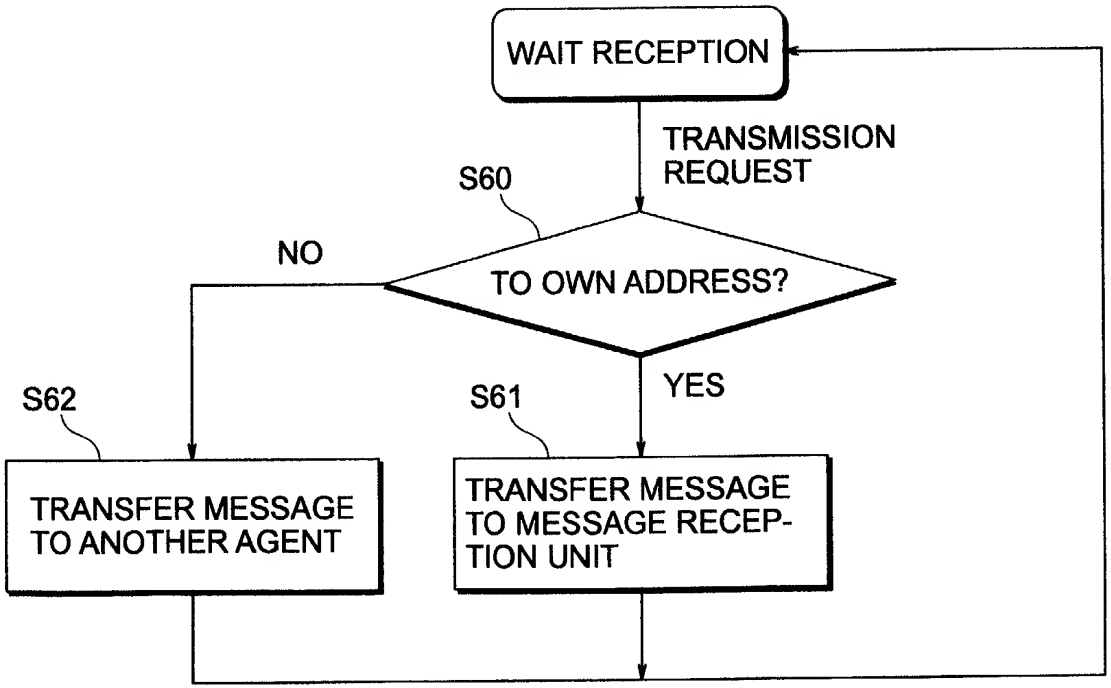
WAIT RECEPTION
TRANSMISSION REQUEST
S60
TO OWN ADDRESS?
NO
YES
S62
S61
TRANSFER MESSAGE TO ANOTHER AGENT
TRANSFER MESSAGE TO MESSAGE RECEP-TION UNIT

FIG. 10A

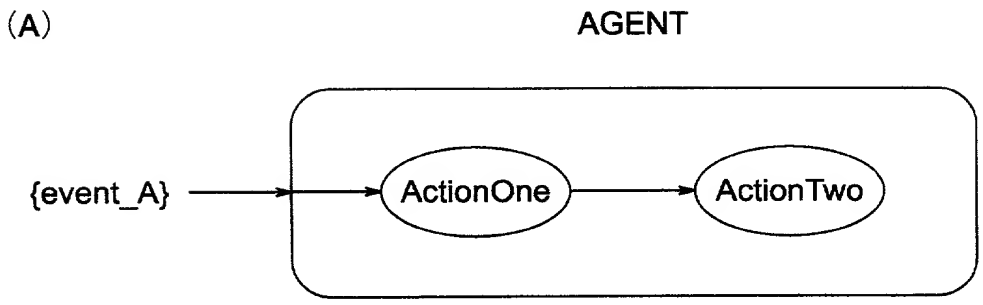

FIG. 10B

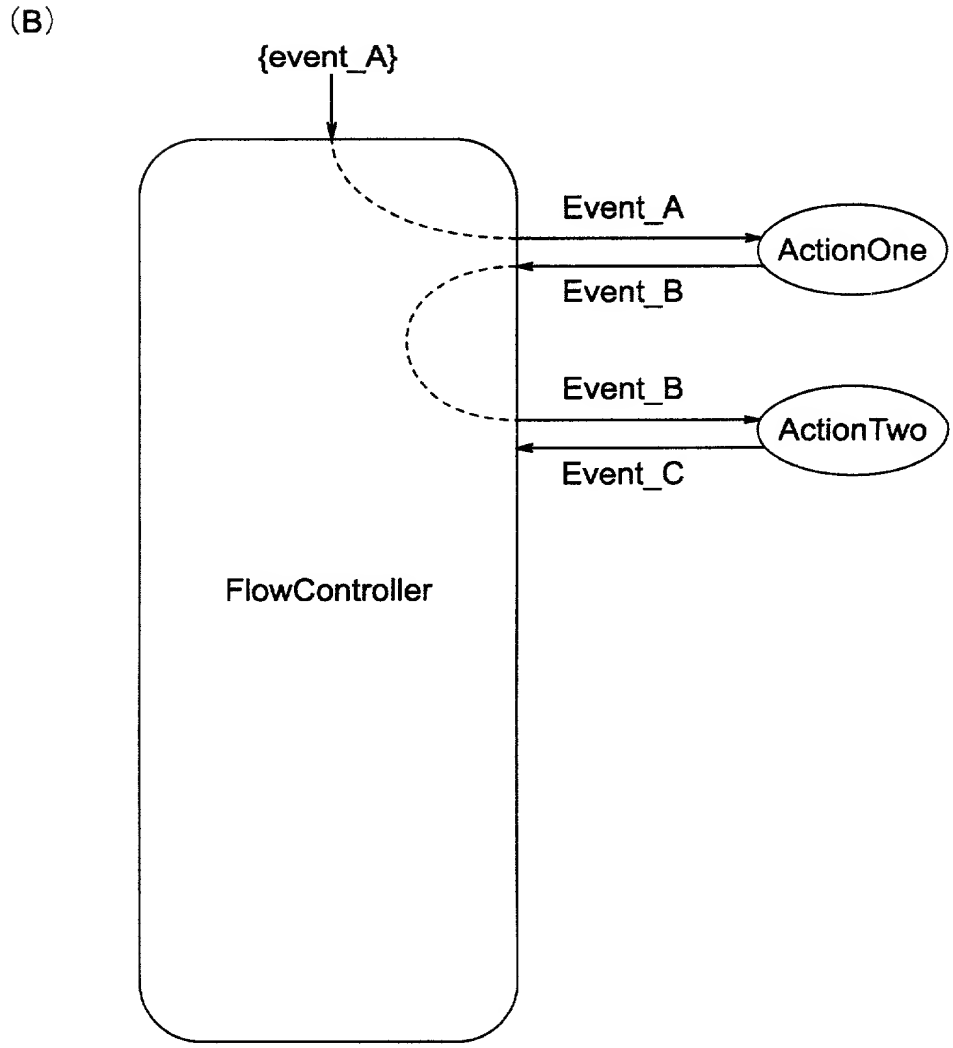

FIG. 11A

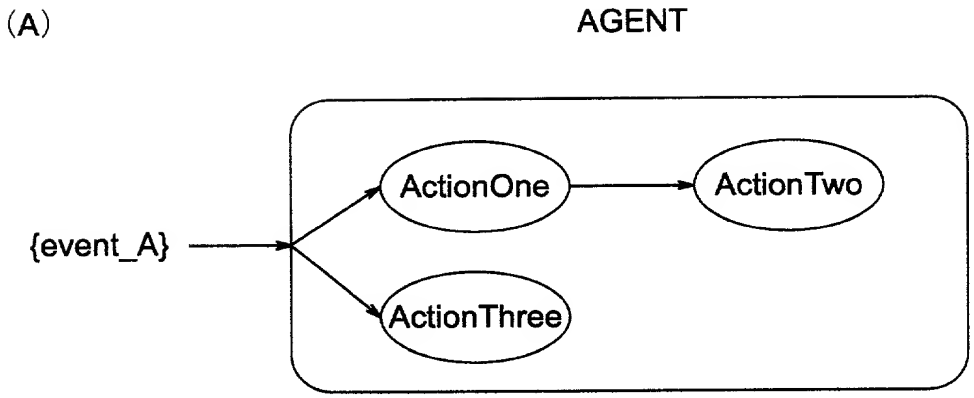

FIG. 11B

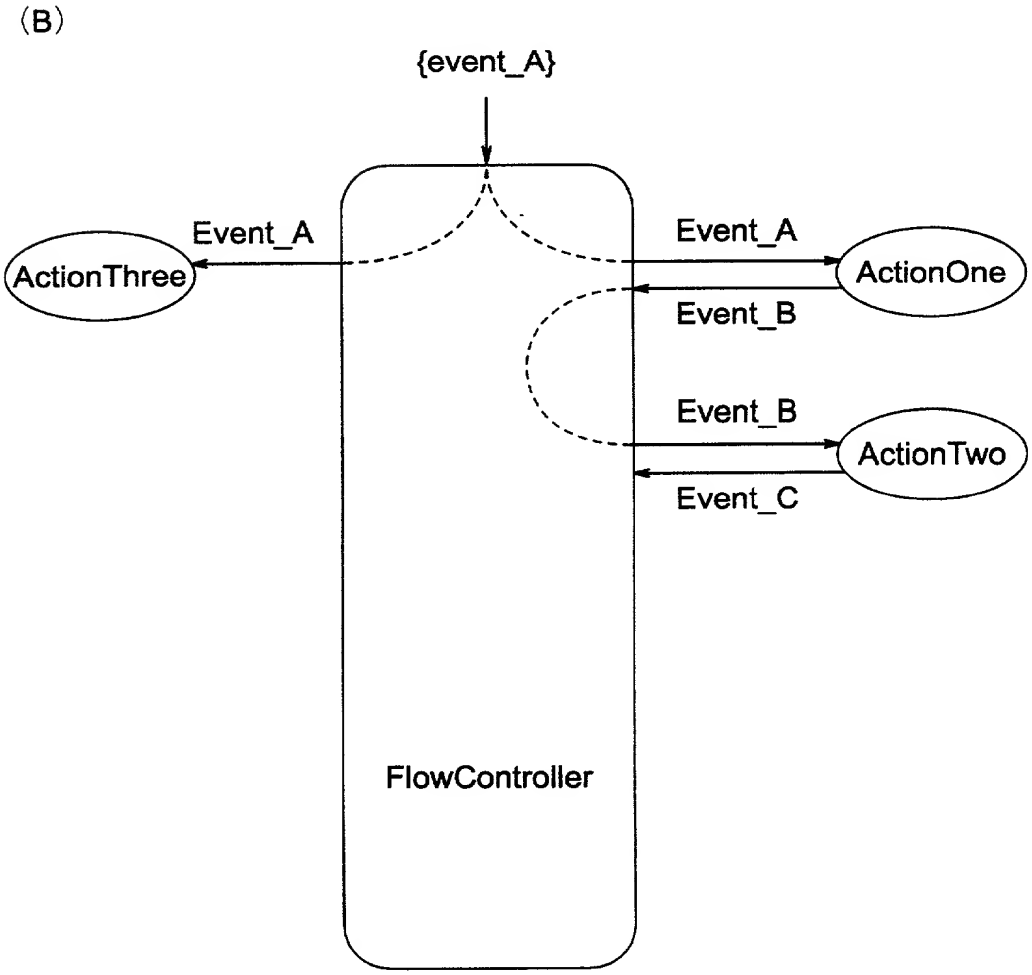

FIG. 12A

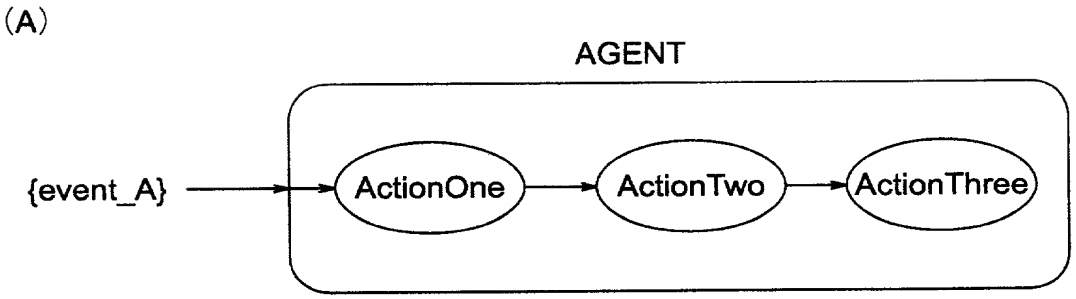

FIG. 12B

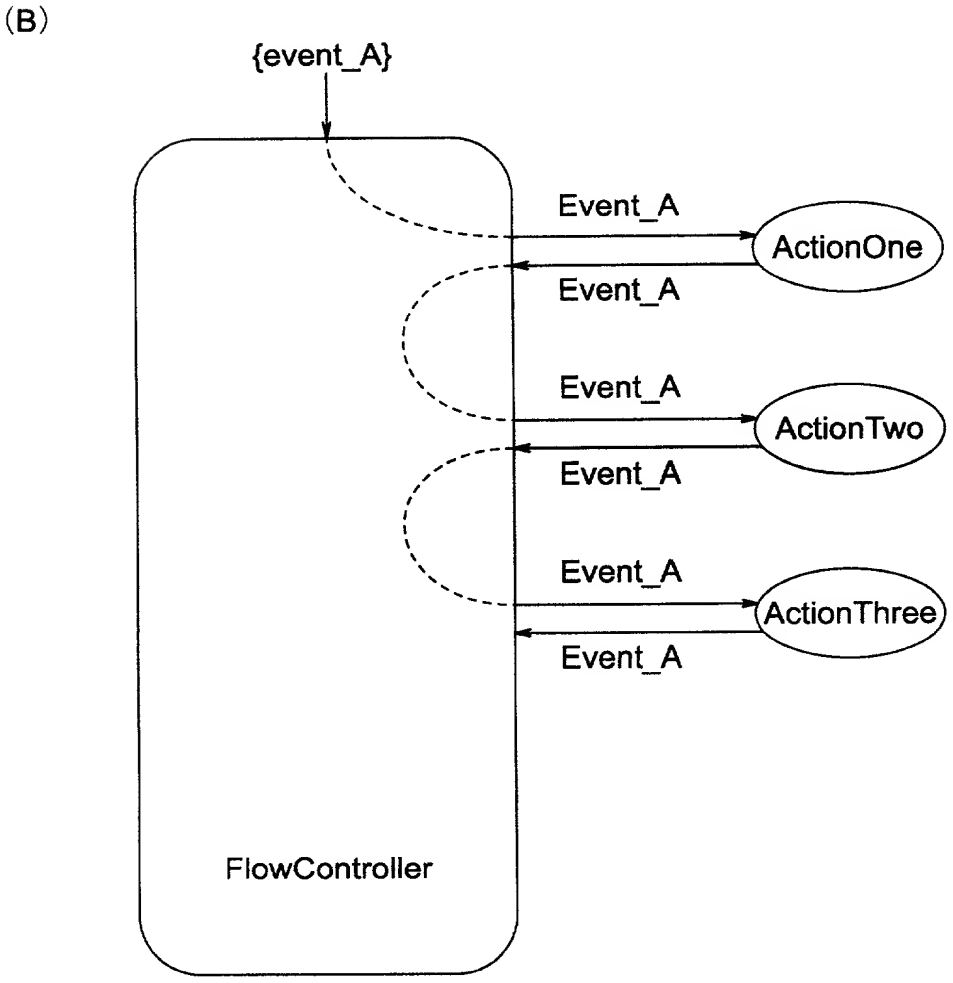

FIG. 13A

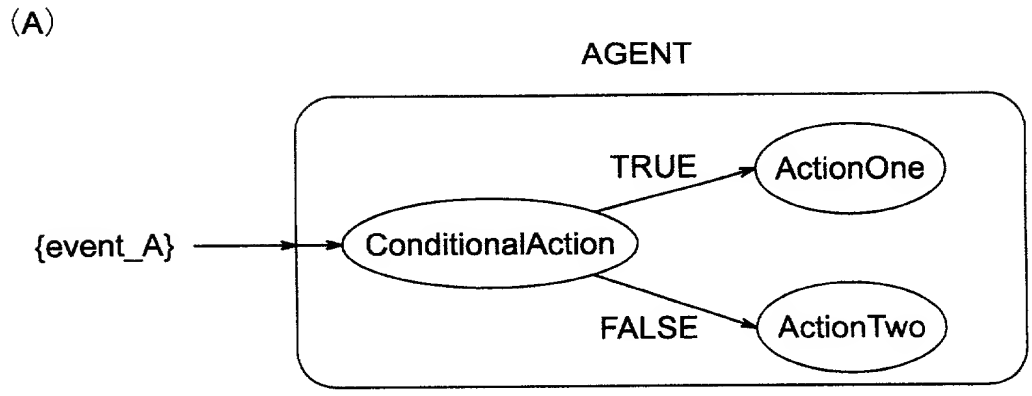

FIG. 13B

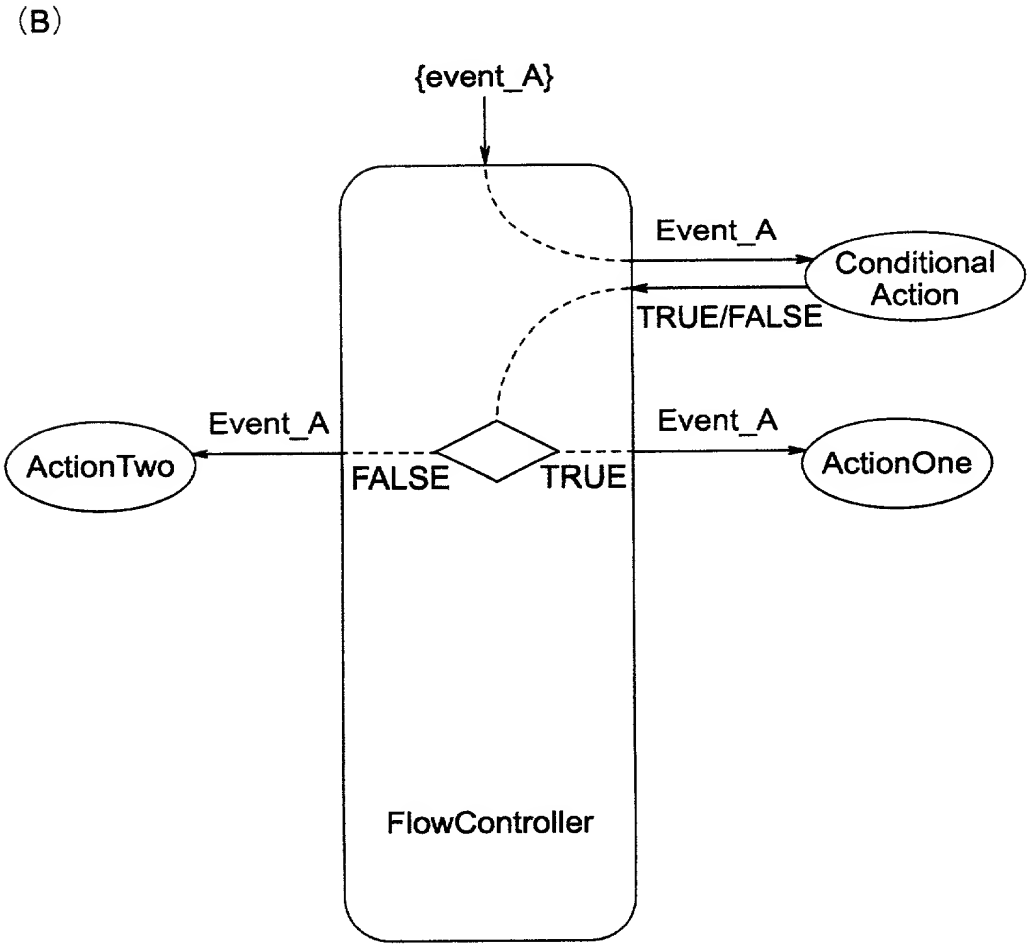

FIG. 14

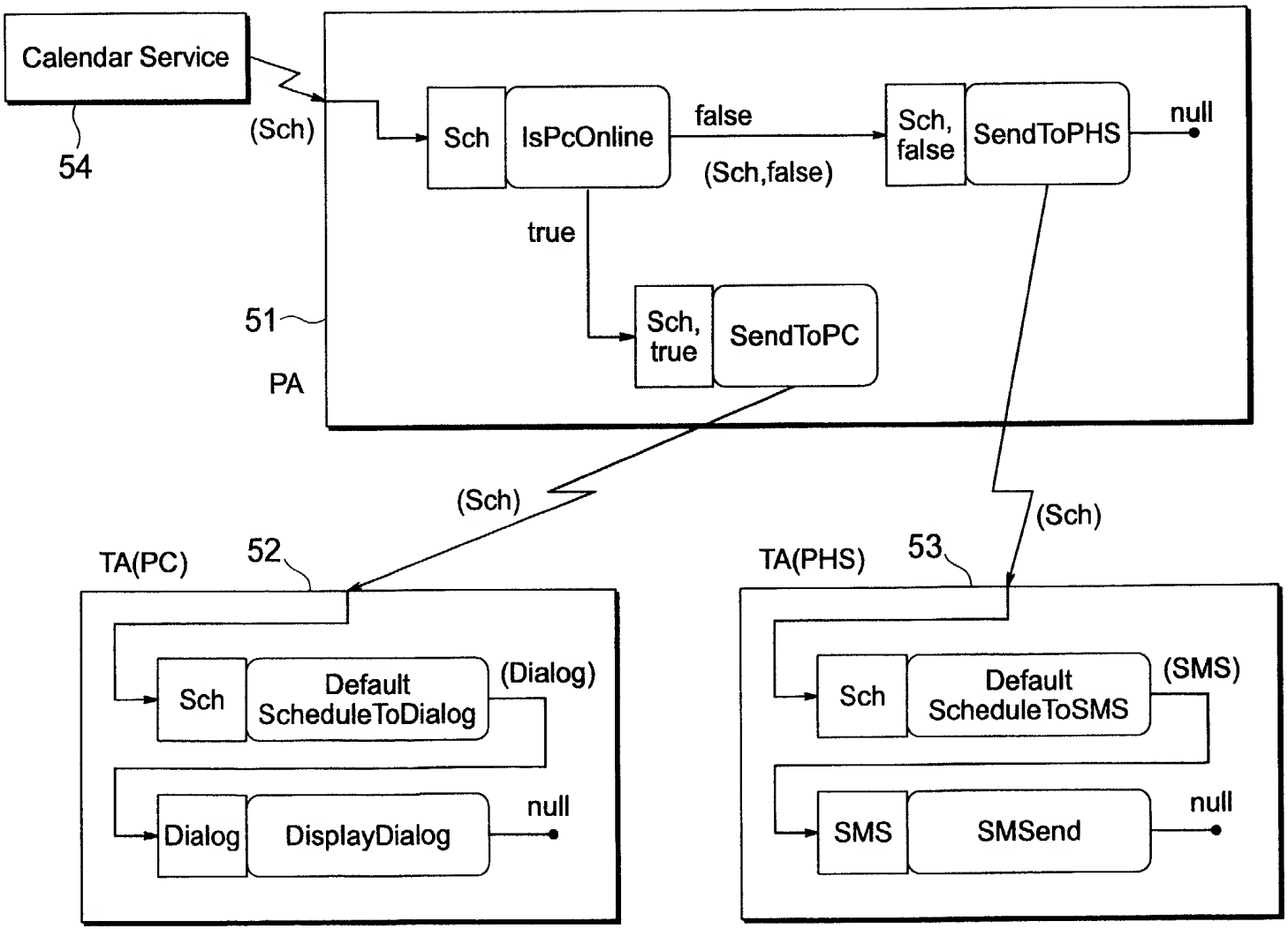
Calendar Service
54
(Sch)
Sch
IsPcOnline
false
(Sch,false)
Sch, false
SendToPHS
null
true
Sch, true
SendToPC
51
PA
(Sch)
(Sch)
TA(PC)
52
TA(PHS)
53
Sch
Default ScheduleToDialog
(Dialog)
Dialog
DisplayDialog
null
Sch
Default ScheduleToSMS
(SMS)
SMS
SMSend
null

FIG. 15

Source object

```
void addEventListener(Listener 1){
    listeners.add(1);
}

void removeEventListener(Listener 1){
    listeners.remove(1);
}

        ⋮

//Runtive phase
while (true) {
    if (status_changed){
        Event ev=..... // prepare event
        for each listener in listener
            listener.action(ev);
    }
}
Vector listeners;
```

Listener object

```
void action(Event ev){
    .....
}

        ⋮

// Initialization phase
Source s= ..... // obtain source rcfs.
s.addBarListeners(this);

        ⋮
```

FIG. 16

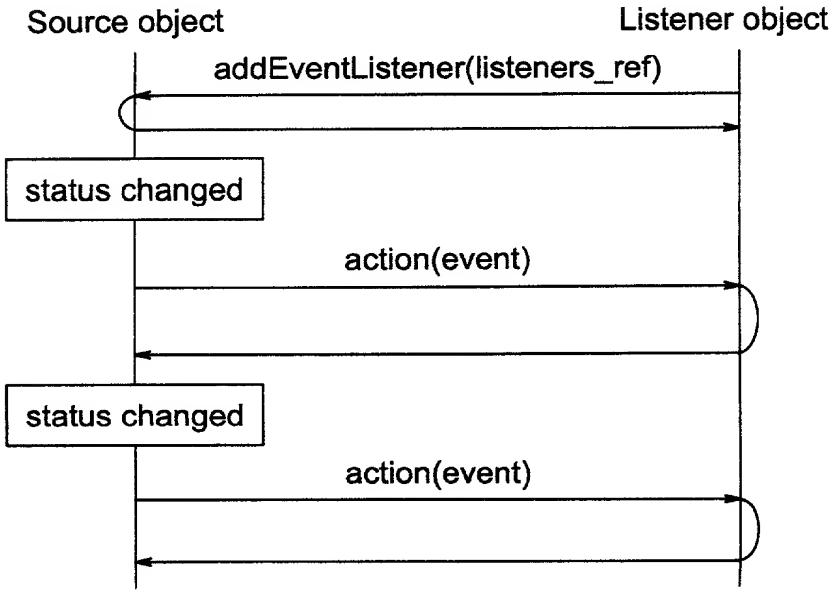

FIG. 17

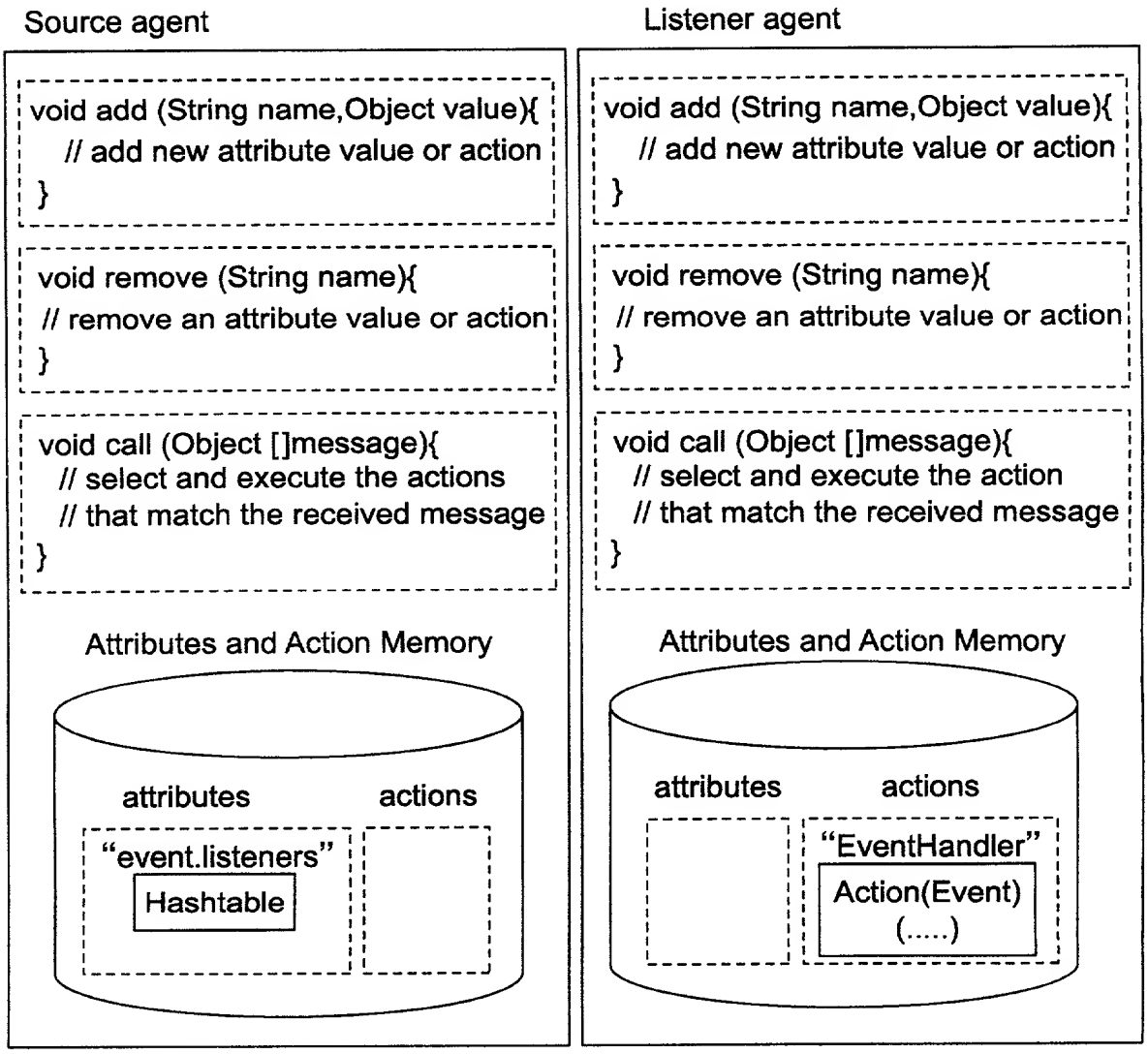
Source agent
Listener agent
void add (String name,Object value){
// add new attribute value or action
}
void remove (String name){
// remove an attribute value or action
}
void call (Object []message){
// select and execute the actions
// that match the received message
}
void add (String name,Object value){
// add new attribute value or action
}
void remove (String name){
// remove an attribute value or action
}
void call (Object []message){
// select and execute the action
// that match the received message
}
Attributes and Action Memory
attributes
actions
"event.listeners"
Hashtable
Attributes and Action Memory
attributes
actions
"EventHandler"
Action(Event)
(.....)

FIG. 18

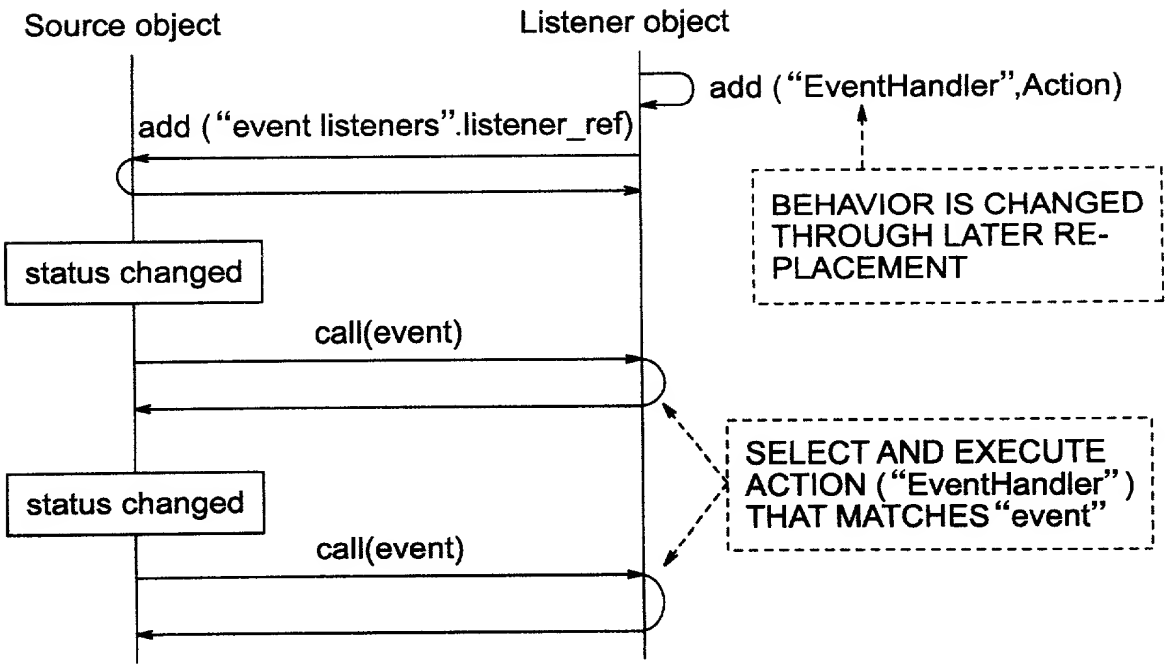
Source object
Listener object
add ("EventHandler",Action)
add ("event listeners".listener_ref)
BEHAVIOR IS CHANGED THROUGH LATER RE-PLACEMENT
status changed
call(event)
SELECT AND EXECUTE ACTION ("EventHandler") THAT MATCHES "event"
status changed
call(event)

FIG. 19

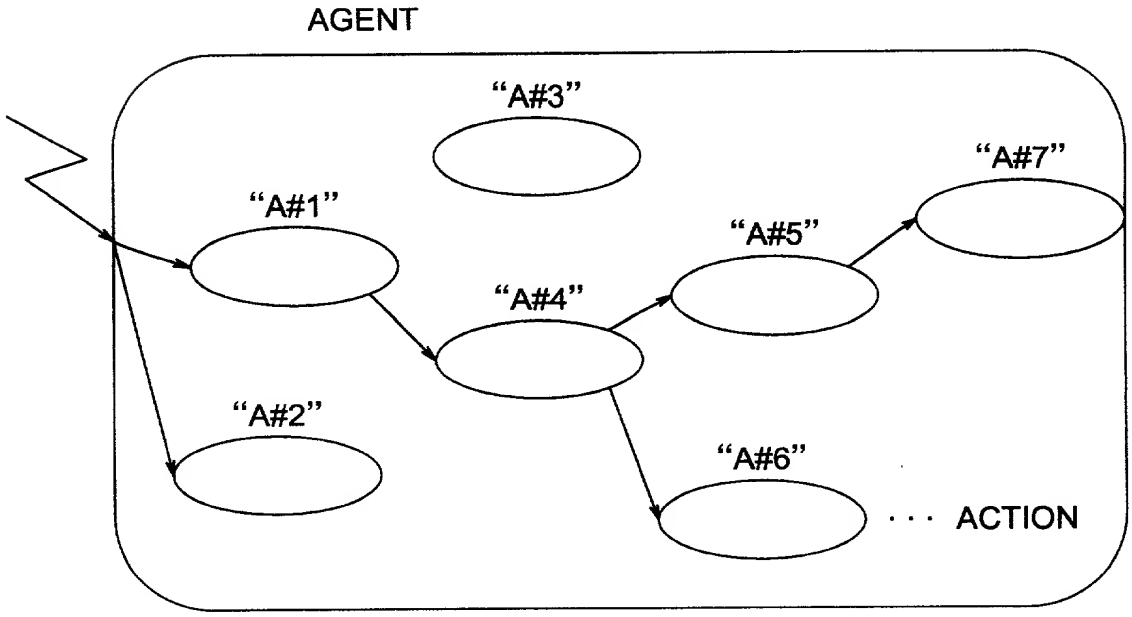
AGENT
"A#3"
"A#7"
"A#1"
"A#5"
"A#4"
"A#2"
"A#6"
· · · ACTION

FIG. 20